445.315 PROPOSITIONAL CALCULUS

Lecturer: Professor Cristian S. Calude, Room 253, ext 5751, e-mail : cristian@cs.auckland.ac.nz

Teaching Assistant: Asat Arslanov, Room 127, ext 7458, email: asat@cs.auckland.ac.nz

The fundamental faith in mathematics comes from the fact that virtually everything is **rigorously proved**. But, what do we mean by a “proof”? According to K. Devlin

> ...[we] mathematicians ... are somewhat schizophrenic when it comes to answering this question. ... we generally feel confident in our ability to tell a sound argument from an invalid one. Moreover, we tend to feel that it really is not an issue of judgment, and that for all their surface brevity, the proofs we construct and publish are, in an absolute sense, genuine *proofs*.

A good way to explain is *to do*. Our choice refers to the simplest way to analyze the mathematical thinking, i.e. by means of the model initiated more than a hundred of years ago by George Boole, currently referred as the *propositional calculus*. We are interested only in the "truth values" of propositions, and again our working hypothesis refers to the simplest case: propositions are only true or false, no other possibility is considered.

It can be argued that even God is bound by logic. If God can lift any weight, then God is expressly prevented to create a weight so heavy that God cannot lift it. But God can do anything that does not involve a logical contradiction. (It seems that Einstein sympathized with this argument.)

We start with a (denumerable) set of *atomic propositions*. Denote by V this set. We add to V a new element, f, referring to it as the *universal false proposition*.

Using the usual propositional operations (disjunction, conjunction, negation, implication, etc.) we can form new propositions starting with the atomic ones. For the simplicity of the presentation we shall work with only one propositional operation – the *implication* (denoted by $\rightarrow$).

We construct the “larger” set of propositions, call it P, defined by means of the following four rules:

1. Every atomic proposition is a proposition, i.e. $V \subset P$.
2. The universal false proposition f belongs to P.
3. If x, y are arbitrary propositions, then $x \rightarrow y$ is a proposition.
4. Every proposition is obtained by rules 1-3.

Using only the implication and f doesn't really make our approach less general; indeed, all propositional operations can be "re-captured" as follows:

$$\text{negation: } \neg x = x \to f,$$

$$\text{disjunction: } x \vee y = (\neg x) \to y,$$

$$\text{conjunction: } x \wedge y = \neg((\neg(x) \vee \neg(y)).$$

How do we know that the above formulas actually work?

To this aim we introduce the notion of valuation. The set of truth values will be denoted by $\{0,1\}$ and we introduce on it the binary operation (called *truth implication*) which models the idea that an implication is false only in case the hypothesis is true but the conclusion is false:

$$\Longrightarrow : \{0,1\} \times \{0,1\} \to \{0,1\},$$

$$m \Longrightarrow n = \max\{1-m, n\}.$$

Here max stands for the usual maximum function. Clearly, $m \Longrightarrow n = 0$ iff $m = 1$ and $n = 0$.

A *valuation* is a function

$$h : V \to \{0, 1\},$$

i.e. a way to assign truth values to atomic propositions. We can extend this valuation to the set of all propositions imposing the following two conditions:[a]

a) $h(f) = 0$,

b) $h(x \to y) = h(x) \Longrightarrow h(y)$, for all $x, y \in P$.

---

[a] Algebraically, h is a morphism; as P is freely generated by V, h is uniquely determined by its values on generators, i.e. on V.

If h is a valuation and $x \in P$, then the proposition x is *true under* h if $h(x) = 1$.

We can now compute the valuations of the negation, disjunction, conjunction, according to a fixed h:

$$h(\neg x) = h(x \to f) = h(x) \Longrightarrow h(f) = h(x) \Longrightarrow 0 = 0 \text{ iff } h(x) = 1,$$

i.e. $h(\neg x) = 1 - h(x)$.

Similarly,

$$h(x \vee y) = h((\neg x) \to y) = h(\neg x) \Longrightarrow h(y)$$

$$= (1-h(x)) \Longrightarrow h(y) = \max\{1-(1-h(x)), h(y)\} = \max\{h(x), h(y)\},$$

$$h(x \wedge y) = h(\neg((\neg(x) \vee \neg(y))))$$

$$= 1 - h(\neg(x) \vee \neg(y)) = 1 - \max\{h(\neg x),$$

$$h(\neg(y)\} = 1 - \max\{1 - h(x), 1 - h(y)\} = \min\{h(x), h(y)\}.$$

The next step is to model the idea of “semantic consequence”: the proposition $a \in P$ can be semantically deduced from the set of premises $X \subset P$ (or X is a semantic model for a) if every valuation which makes **all** premises in X true, makes true a as well.

Formally,

$$X \models a$$

if for every valuation h such that $h(x) = 1$, for all $x \in X$, one has $h(a) = 1$.

Remark. It is interesting to note that in LaTeX the symbol $\models$ is written `\models`. In case $X = \emptyset$ we simple write $\models a$.

Example 0.1. *The following relations are true:*

1. $\models a \to (b \to a)$, *for all* $a, b \in P$.
2. $\models (a \to (b \to c)) \to ((a \to b) \to (a \to c))$, *for all* $a, b, c \in P$.
3. $\models ((a \to f) \to f) \to a$, *for every* $a \in P$.
4. $\{a\} \models b \to a$, *for all* $a, b \in P$.

For instance, for the last relation we have to show that for every valuation h such that $h(a) = 1$ one has $h(b \to a) = 1$. Indeed, $h(b \to a) = h(b) \Longrightarrow h(a) = h(b) \Longrightarrow 1 = 1$.

A special category of propositions is formed by "universal true propositions", i.e. propositions which are true with respect to all possible valuations. We call them *tautologies*. The propositions $a \to (b \to a), (a \to (b \to c)) \to ((a \to b) \to (a \to c)), ((a \to f) \to f) \to a$ are all tautologies.

Of course, not all propositions are tautologies; the extreme example is a proposition that is never true under any valuation, for instance, $a \wedge \neg a$.

Remark. From a strict formal point of view, $a \wedge \neg a$ is *not* a proposition in P, as $\wedge, \neg$ are not admissible operators. However, we shall use the formula $a \wedge \neg a$ as an abbreviation for the proposition

$$(((a \to f) \to f) \to ((a \to f) \to f)) \to f.$$

Example 0.2. *The following propositions are tautologies:*

1. $a \to a$, *(identity principle)*,
2. $a \vee \neg a$, *(tertium non datur)*,
3. $(a \wedge (a \to b)) \to b$, *(modus ponens)*,
4. $((a \to b) \wedge (a \to \neg b)) \to \neg a$ *(negation principle)*,
5. $a \to (a \vee b)$, *(disjunction principle)*.

Several natural questions can be asked, for instance:

- Is there a compact way to “describe” all and only all tautologies?
- What is the “structure” of the set of tautologies?
- Is it possible to “algorithmically” recognize a tautology? Is this a feasible task?

There are several possibilities to describe a set of propositions, specifically the set of tautologies. For our purpose the most interesting one is the deductive approach. The prototype of a deductive science is Euclid's geometry, developed around 300 B.C. The major step undertook by Euclid and his predecessors in Greece was to organise these facts into a deductive science or axiomatic-deductive geometry.

The truth or falsity of most propositions in geometry cannot be seen directly from their meanings. The *axioms*, however, are special propositions whose truths are immediately recognized from their meanings; in fact, for a long period, this was the major criterion to select axioms (we shall return to this problem later). Starting with axioms, by a series of logical steps that we accept as propagating truth forward, we construct a “proof” by which we can arrive at the truth of other propositions – called “theorems”.

Do we have a “solid” basis for recognizing the rules that allow us to propagate truth forward? This is a very delicate problem. For a long period Euclid’s theory was considered the prototype of a perfect theory. However, in the nineteenth century people revealed flaws in Euclid’s proofs, making essential use of “illustrated” figures. False “theorems” were discovered: “proofs” read just Euclid’s proofs, except that some figures were “fudged” a little bit.

A fundamental change of view point has to be adopted: *deductions should be possible to carry out without reference to meanings.* In Euclid's case, proofs should read correctly with nonsense words substituted for "point", "line", "plane". The logical principles which mediate the steps in proofs should be stated in advance as *rules of inference.* So, the meanings of none of the words need to be considered in constructing proofs; the quality of being a valid proof depends then only on the form of the sentences.

A valid proof has to be impersonal: whenever an alleged proof is submitted to a person who has previously been told the specifications of the system she/he should be able to *check* the proposed proof and decide whether it actually is a proof or not. No extra imagination or judgment is needed. In other terms, checking the validity of an alleged proof may be done by computer; this infinite class of yes-or-no questions is *algorithmically decidable*.

This explains why the working mathematicians may differ (and, indeed, they do) in their views of what constitutes a rigorous proof, in spite of the fact that all of them believe in rigorous proofs.

We shall now explain an axiomatic system for tautologies. We first "isolate" a few tautologies (called *axioms*) and a "deduction principle" with the aim of "deriving" all and only all tautologies. The axioms will be the first three tautologies:

A1. $a \to (b \to a)$, for all $a, b \in P$.

A2. $(a \to (b \to c)) \to ((a \to b) \to (a \to c))$, for all $a, b, c \in P$.

A3. $((a \to f) \to f) \to a$, for every $a \in P$.

As a deduction rule we make use of the most fundamental principle (called *Modus Ponens*), modeling the following inference: if one proves y from x, and x has a proof, then y has a proof.

Modus Ponens : *For all* $x, y \in P$, *if* x *and* $x \to y$, *then* y.

We have got a “formal system”. Within it we can discuss about “(formal) **proofs**”.

Remark. To distinguish between “proofs within our system” and “proofs outside the system”, the first ones will be written in bold characters.

Informally, a **proof** is just a finite sequence of propositions such that every element in the sequence is an axiom or can be deduced from propositions already in the sequence by *Modus Ponens*. Sometimes our **proofs** make use of extra hypotheses X; they will be called X**-proofs**.

Let X be a set of propositions and $a \in P$. An X**-proof** for a (i.e. a **proof** (within the system) of a from the set X of premises) is a sequence

$$x_1, x_2, \ldots, x_n$$

of elements in P such that $x_n = a$ and for all $1 \leq i \leq n$ one has:

- $x_i \in X$ or,
- x_i is an axiom, or
- there exist $1 \leq k, l < i$ such that $x_l = x_k \to x_i$ (i.e. x_i can be deduced from x_k and x_l via *Modus Ponens*).

The proposition a is called an X**-theorem** in this case.

Sometimes we write:

$$X \vdash a,$$

or simply

$$\vdash a,$$

in case $X = \emptyset$.

Example 0.3. *For all $a, b, c \in P$, the following relations are true:*

1. $\vdash a \to (b \to a)$,
2. $\vdash a \to a$,
3. $\{b\} \vdash a \to b$,
4. $\{a \to (c \to b), a \to c\} \vdash a \to b$.

Proof. The first proposition, $a \to (b \to a)$, is an axiom. For $a \to a$ we can write the following **proof**:

1. $a \to ((b \to a) \to a)) \to ((a \to (b \to a)) \to (a \to a))$ (by A2.)
2. $a \to ((b \to a) \to a)$ (by A1.)
3. $(a \to (b \to a)) \to (a \to a)$ (by *Modus Ponens* from 2. and 1.)
4. $a \to (b \to a)$ (by A1.)
5. $a \to a$ (by *Modus Ponens*, from 4. and 3.).

The following sequence represents an $\{b\}$-**proof**:

1. $b \rightarrow (a \rightarrow b)$
2. b
3. $a \rightarrow b$.

Finally, the following sequence represents an $\{a \to (c \to b), a \to c\}$**-proof**:

1. $(a \to (c \to b)) \to ((a \to c) \to (a \to b))$
2. $a \to (c \to b)$
3. $(a \to c) \to (a \to b)$.
4. $a \to c$
5. $a \to b$.

The following theorem, due to Herbrand, makes explicit the relation between the implication ($\rightarrow$), as an inner operator of the system, and the syntactical derivation ($\vdash$), the external deduction.

Deduction Theorem. *Let $X \subset P$, and $a, b \in P$. The following statements are equivalent:*

a) $X \vdash a \rightarrow b$,

b) $X \cup \{a\} \vdash b$.

Proof. For the direct implication let $x_1, x_2, \ldots, x_n$ be an X**-proof** for $a \to b$. Then $x_1, x_2, \ldots, x_n, a, b$ is an $X \cup \{a\}$**-proof** for b.

We are proving the converse implication. Let $x_1, x_2, \ldots, x_n$ be an $X \cup \{a\}$**-proof** for b. We prove, by induction on i, that

$$X \vdash a \to x_i.$$

For $i = n$ we get the desired conclusion:

$$X \vdash a \to b,$$

as $x_n = b$.

There are four possible cases to be discussed:

- $x_i \in X$: since $\{x_i\} \vdash a \to x_i$ one deduces $X \vdash a \to x_i$, as $x_i \in X$.
- $x_i = a$: since $\vdash a \to a$, by Example 0.3, $X \vdash a \to a$.
- x_i is an axiom: one has $\vdash x_i$, so $\{x_i\} \vdash a \to x_i$, i.e. $X \vdash a \to x_i$.
- $x_k = x_j \to x_i, j, k < i$: by hypothesis, $X \vdash a \to x_j$ and $X \vdash a \to x_k$. By virtue of Example 0.3,

$$\{a \to x_j, a \to (x_j \to x_i)\} \vdash a \to x_i,$$

so $X \vdash a \to x_i$.

Remark. There is a simpler, semantical analogue of the Deduction Theorem. It reads as following: *Let* $X \subset P$, *and* $a, b \in P$. *The following statements are equivalent:*

a) $X \models a \rightarrow b$,

b) $X \cup \{a\} \models b$.

Here are two more examples of X**-proofs**:

Example 0.4.

1. $\{f\} \vdash a$, *for all* $a \in P$,
2. $\{a, \neg a\} \vdash b$, *for all* $a, b \in P$.

Proof. Here is an $\{f\}$**-proof** for a:

1. f
2. $f \to ((a \to f) \to f)$
3. $(a \to f) \to f$
4. $((a \to f) \to f) \to a$
5. a.

Starting with the sequence

a. a

b. $a \to f$

c. f

and the above $\{f\}$**-proof**, i.e. steps 1.,2.,3.,4.,5. we get an arbitrary proposition b.

Adequacy Problems

It's time to look critically at our system. Is it adequate? *Soundness* is the first required property, as we are interested to "describe" only tautologies. Specifically, the question reads: Is any **theorem** a tautology? A negative answer would ruin the whole construction.

We start proving that every X**-theorem** can be semantically deduced from X:

Proposition. *For all* $X \subset P, a \in P$, *if*

$$X \vdash a,$$

then

$$X \models a.$$

Proof. We use the definition of an X**-proof** inductively, noticing that all axioms are tautologies and *Modus Ponens* is an invariant rule.

To be really successful we need to prove the completeness of our system. The following proof will give some more insight on the nature of tautologies.

A set $X \subset P$ is called *consistent* if it is free of contradictions; formally, if

$$X \nvdash f.$$

Example 0.5. *The empty set is consistent. The set $\{a, \neg a\}$ is not consistent.*

Proof. Indeed, if $\vdash f$, then, $\models f$, which is absurd as for every valuation h one has $h(f) = 0$. By Example 0.4, $\{a, \neg a\} \vdash f$.

We shall prove now two technical results that are motivated by a typical algebraic construction: the embedding of a structure in a maximal structure of the same type.

Lemma. *The union of an increasing sequence (under set-theoretical inclusion) of consistent sets is still a consistent set.*

Proof. Let

$$B_1 \subset B_2 \subset \cdots \subset B_n \subset B_{n+1} \subset \cdots$$

be an increasing sequence of consistent sets and put

$$B = \bigcup_{n \geq 1} B_n.$$

If, by absurd, B is not consistent, then $B \vdash f$, i.e. there exists a finite set $X \subset B$ such that $X \vdash f$. So, $X \subset B_m$, for some natural $m \geq 1$ (as the sequence $(B_i)_{i \geq 1}$) is increasing). This contradicts the consistency of B_m.

A set $X \subset P$ is called *maximal consistent* if for every element $a \notin X$ the set $X \cup \{a\}$ is not consistent.

Proposition. Every consistent set can be embedded into a maximal consistent set.

Proof. The usual way to prove such a result is to invoke Zorn's Lemma. Let X be a consistent subset of P. The set $\Gamma = \{T \subset P | X \subset T, T \nvdash f\}$ is non-empty $(X \in \Gamma)$. More, every chain in Γ has an upper bound: if $\{T_\alpha | \alpha \in \Lambda\}$ is a totally ordered family of elements of Γ and $T = \bigcup_{\alpha \in \Lambda} T_\alpha$, then $X \subset T \subset P, T \nvdash f$ (as, if f is provable from T means that f is provable from a finite subset of T).

Use now Zorn's Lemma (*if S is a partially ordered set in which each chain has an upper bound, then S contains a maximal element*) to assert the *existence* of a maximal element in Γ. In general, no claim of constructivity can be made for such a reasoning

We proceed *constructively*, i.e. introducing an enumeration technique usually referred to as a *gödelization*. Assume that we have an one-one enumeration $v_i, i = 2, 3, \ldots$ of all atomic propositions.

Then, we can construct an one-one function $g : P \rightarrow \mathbf{N}$ as follows. Assume that our enumeration $\{v_i\}$ was "computable", in the sense that there exists an *algorithm* computing v_i when presented i. Then, the function g is itself computable and given $i \in g(P)$ we can effectively discover the (unique) proposition $x \in P$ such that $g(x) = i$.

Put

$$g(f) = 0,$$

$$g(v_i) = 2i - 1, i \geq 1,$$

$$g(x \rightarrow y) = 2^{g(x)}.3^{g(y)}.$$

For instance,,

$$g(v_1 \rightarrow (v_2 \rightarrow v_1)) = 2^{g(v_1)}.3^{g(v_2 \rightarrow v_1))}$$

$$= 2^1.3^{2^{g(v_2)}3^{g(v_1)}} = 2.3^{2^3.3^1} = 2.3^{24}.$$

Let $X \subset P$ be a fixed consistent set. Using the above function g we define the following sequence of sets of propositions:

$$B_0 = X,$$

$$\begin{aligned} B_{n+1} &= B_n \cup g^{-1}\{n\}, && \text{if } B_n \cup g^{-1}\{n\} \text{ is consistent,} \\ &= B_n, && \text{otherwise.} \end{aligned}$$

Clearly, the sequence B_n is increasing and each B_n is consistent. So, $B = \bigcup_{n \geq 0} B_n$ is consistent as well. Since $B_0 = X \subset B$ the only fact it remains to be proven is the maximal consistency of B. Let $B \subset Y \subset P$ be such that $B \neq Y$, i.e. there exists a proposition

$$g^{-1}(n) \in Y, g^{-1}(n) \notin B.$$

From the relation $g^{-1}(n) \notin B$ and the construction of B_{n+1} it follows that $g^{-1}(n) \notin B_{n+1}$ – because $B_n \cup \{g^{-1}(n)\}$ is not consistent, i.e.

$$B_n \cup \{g^{-1}(n)\} \vdash f.$$

But, $B_n \subset Y$, $g^{-1}(n) \in Y$, so $B_n \cup \{g^{-1}(n)\} \subset Y$ and $Y \vdash f$, saying that Y is not consistent.

Maximal consistent sets are "fixed-points" of the operator generating **theorems** and they obey the Bivalence Principle.

Proposition. *If $X \subset P$ is a maximal consistent set, then*

1. $\{a \in P | X \vdash a\} = X$,
2. *for every $a \in P$, one and only one of the following relations is true: $a \in X$ or $\neg a \in X$.*

Proof. 1) Clearly, $X \subset \{a \in P | X \vdash a\}$. But

$$f \notin \{a \in P | X \vdash a\} = \{a \in P | \{b \in P | X \vdash b\} \vdash a\},$$

so $\{a \in P | X \vdash a\}$ is consistent. In view of the inclusion $X \subset \{a \in P | X \vdash a\}$ we can make use of the maximality to derive the equality.

2) If $a \notin X$, then $X \cup \{a\}$ is not consistent, i.e.

$$X \cup \{a\} \vdash f.$$

We use now the Deduction Theorem to get

$$X \vdash a \to f = \neg a.$$

By virtue of 1), $\neg a \in X$. If $\neg a \notin X$, then $X \cup \{\neg a\}$ is not consistent,

$$X \cup \{\neg a\} \vdash f.$$

By the Deduction Theorem

$$X \vdash \neg a \to f = (a \to f) \to f = \neg\neg a.$$

By the axiom A3.

$$\neg\neg a \to a,$$

so $X \vdash a$, i.e. $a \in X$ by virtue of maximality. Of course, it is not the case that both a and $\neg a$ are in X, as X is consistent.

The next result shows that every maximal consistent set of propositions X has a model.

Proposition. *Let X be a maximal consistent set. Then, c_X, the characteristic function of X (with respect to P) is a valuation making true all propositions in X.*

Proof. Recall that $c_X : P \to \{0, 1\}$, $c_X(a) = 1$ iff $x \in X$. As X is maximal consistent, $X \nvdash f$, so $f \notin X$, i.e. $c_X(f) = 0$.

Take now $a, b \in P$. We shall prove that

$$c_X(a \to b) = c_X(a) \Longrightarrow c_X(b).$$

There are three cases to be analysed.

1. *If* $b \in X$, then $c_X(b) = 1$, so $c_X(a) \Longrightarrow c_X(b) = c_X(a) \Longrightarrow 1 = 1$. One has to prove that $c_X(a \to b) = 1$, i.e. $a \to b \in X$. We know that X is maximal consistent, so a fixed-point:

$$X = \{c \in P | X \vdash c\}.$$

By axiom A1., $\vdash b \to (a \to b)$, so

$$X \vdash b \to (a \to b).$$

By hypothesis, $b \in X$, so $X \vdash b$ and (by *Modus Ponens*) $X \vdash a \to b$. Again, by maximality, $a \to b \in X$.

2. *If* $a \notin X$, then $c_X(a) = 0$, so $c_X(a) \Longrightarrow c_X(b) = 0 \Longrightarrow c_X(b) = 1$. So, we have to prove again the relation $c_X(a \to b) = 1$. But $a \notin X$ implies, $\neg a \in X$. Using the Deduction Theorem (to $\{a, \neg a\} \vdash b$) we get

$$\{\neg a\} \vdash a \to b.$$

Since $\neg a \in X$, $X \vdash \neg a$, so $X \vdash a \to b$, i.e. $a \to b \in X$.

3. *If* $a \in X$ *and* $b \notin X$, then $c_X(a) = 1, c_X(b) = 0$, so $c_X(a) \Longrightarrow c_X(b) = 0$. The relation $a \to b \notin X$ remains to be proven. Indeed, if $a \to b \in X, a \in X$, then $b \in X$, a contradiction.

Corollary. *If X is consistent, then there exists a valuation $h : P \to \{0,1\}$ such that $h(a) = 1$, for all $a \in X$.*

Proof. Embed the given consistent set into a maximal consistent set and then use the proposition on models.

We are now able to prove

Post's Completeness Theorem. *For every set of propositions X and every proposition a,*

$$X \models a \text{ iff } X \vdash a.$$

Proof. Only the direct implication has to be proven. Assume that $X \models a$. We shall prove that $X \cup \{\neg a\}$ is not consistent. If it were consistent, then we would have, a valuation h such that $h(b) = 1$, for all $b \in X \cup \{\neg a\}$, i.e.

$$h(\neg a) = 1,$$

and

$$h(b) = 1, \text{ for all } b \in X.$$

From $h(\neg a) = 1$ one deduces $h(a) = 0$, so we have contradicted the hypothesis $X \models a$.

From the inconsistency of the set $X \cup \{\neg a\}$ we deduce

$$X \cup \{\neg a\} \vdash f.$$

Use again the Deduction Theorem

$$X \vdash \neg a \rightarrow f = \neg\neg a,$$

and the axiom A3. to get $X \vdash a$.

Problem: Does it mean that actually we have got an $X \cup \{\neg a\}$ **-proof** for f or only an assertion telling that such a **proof** does exist?

Taking $X = \emptyset$ get

For every proposition a,

$$\models a \textit{ iff } \vdash a.$$

Digression: *Three or many valued logics.* The proposition

$$(\neg a \rightarrow a) \rightarrow a$$

is clear a tautology (it seems to be discovered by Clavius, 1600 A.C.). If we switch the underlying logic, from binary, to, say ternary, we loose the tautological property. Indeed, assume we work with the ternary logic in which the truth values are $0, \frac{1}{2}$ (uncertain)$, 1$. The implication valuation will use the same formula, i.e. $m \Longrightarrow n = \max\{1 - m, n\}$. This means that $\neg\frac{1}{2} = \frac{1}{2}$ and $\frac{1}{2} \Longrightarrow \frac{1}{2} = \frac{1}{2}$. If the truth value of a is uncertain, then the proposition $(\neg a \rightarrow a) \rightarrow a$ is also uncertain.

The Structure of Tautologies

One reason why Boolean algebras are relevant to logic is that propositional operators have properties similar to Boolean operations. Based on this analogy we cannot ask questions such as “What is a proof?” or “How can we prove?”; instead, we can study the inner “structure” of provable propositions, that is, via the Completeness Theorem, of tautologies.

To every valuation h we associate an equivalence relation $\stackrel{h}{\sim}$ defined on P as follows:

$$a \stackrel{h}{\sim} b \text{ iff } h(a) = h(b).$$

In fact, $\stackrel{h}{\sim}$ is more than an equivalence relation, it is a *congruence*, in the sense that $\stackrel{h}{\sim}$ is compatible with the algebraic structure of P: If $a \stackrel{h}{\sim} b$ and $a' \stackrel{h}{\sim} b'$, then $a \to a' \stackrel{h}{\sim} b \to b'$. Indeed, from $h(a) = h(b), h(a') = h(b')$ we deduce

$$h(a \to a') = h(a) \Longrightarrow h(a') = h(b) \Longrightarrow h(b') = h(b \to b').$$

The intersection of congruences $\overset{h}{\sim}$, when h runs over all valuations,

$$\sim \; = \bigcap_{h \text{ valuation}} \overset{h}{\sim},$$

is still a congruence on P. The equivalence class of an element $a \in P$ is

$$[a] = \{b \in P | a \overset{h}{\sim} b, \text{ for every valuation } h\}.$$

The family of all equivalence classes, $P/_{\sim} = \{[a]\}_{a\in P}$ can be endoweded, in a natural way, with the following Boolean operations:[a]

$$\neg[a] = [\neg a],$$

$$[a] \vee [b] = [\neg a \to b],$$

$$[a] \wedge [b] = [\neg((\neg a) \vee (\neg b))].$$

---

[a] Recall that $\neg a$ is an abbreviation for the proposition $a \to f$.

These definitions are correct, i.e. they do not depend upon the chosen “names” for classes. For instance, if $a \sim b$, then $h(a) = h(b)$, for every valuation h, so

$$h(\neg a) = h(a \to f) = h(a) \Longrightarrow h(f) =$$

$$h(a) \Longrightarrow 0 = h(b) \Longrightarrow 0 = h(b \to f) = h(\neg b),$$

meaning that $[\neg a] = [\neg b]$. More, $(P/_{\sim}, \vee, \wedge, \neg)$ is a Boolean algebra (called *Lindenbaum algebra)*.

The following identities are satisfied for all $u, v, w \in P/_{\sim}$:

$$u \wedge v = v \wedge u,\ u \vee v = v \vee u,$$

$$u \wedge (v \wedge w) = (u \wedge v) \wedge w,\ u \vee (v \vee w) = (u \vee v) \vee w,$$

$$u \wedge (u \vee v) = u, u \vee (u \wedge v) = u,$$

$$u \wedge (v \vee w) = (u \wedge v) \vee (u \wedge w),\ u \vee (v \wedge w) = (u \vee v) \wedge (u \vee w),$$

$$(u \wedge \neg u) \vee v = v,\ (u \vee \neg u) \wedge v = v.$$

The distinguished element “1” of this Boolean algebra is

$$\mathbf{Taut} = [a] \vee \neg[a] = [a] \vee [\neg a] = [a \vee \neg a],$$

and, as $h(a \vee \neg a) = 1$, for all $a \in P$, coincides with the set of all tautologies. Recall that in every Boolean algebra the expression $x \vee \neg x$ does not depend upon the actual value of x; this is the element “1”, or the maximal element of the algebra.

We make one more step further in our generalization: Consider an arbitrary Boolean algebra B endowed with the operations $\vee, \wedge, \neg$, whose elements are identified with the propositions of some mathematical theory. Assume $F \subset B$ corresponds to the set of *provable* propositions (*theorems*).

The common mathematical experience motivates the following two statements;

- If $s, t \in F$, then s **and** $t \in F$.
- If $s \in F$ and $t \in B$, then s **or** $t \in F$.

The above two properties are similar to properties defining the notion of *Boolean filter*.

Is this only a superficial analogy? The argument for a negative answer is presented in the following

Theorem. *Let F be a subset of the Boolean algebra B. Then, the following two assertions are equivalent:*

1. *The set F is a filter.*[a]

2. *One has:*

  a. $1 \in F$.

  b. *If $x \in F$ and $x \to y \in F$, then $y \in F$.*

---

[a] That is, 1. $1 \in F$, 2. For all $x, y \in F, x \wedge y \in F$, 3. For all $x \in F, y \in B, x \vee y \in F$.

Proof. For the direct implication we assume that $x \in F$ and $x \to y \in F$. In view of the second property of a filter, if $p, q \in F$, then $p \wedge q \in F$. A simple computation shows that $x \wedge (x \to y) = x \wedge (\neg x \vee y) = (x \wedge \neg x) \vee (x \wedge y) \in F$, i.e. $x \wedge y \in F$. Finally, $y = y \vee (x \wedge y)$ is in F as $x \wedge y \in F$; we have used the third property of a filter.

Conversely, let $x, y \in F$. We have: $x = x \wedge (x \vee y)$, so

$$\neg x = \neg x \vee \neg(x \vee y) = \neg x \vee (\neg x \wedge \neg y).$$

Consequently,

$$1 = x \vee \neg x = x \vee (\neg x \vee (\neg x \wedge \neg y)) = (x \vee (\neg x \wedge \neg y)) \vee \neg x \in F.$$

But $x \in F$ and $1 \in F$, so $x \vee (\neg x \wedge \neg y) \in F$ and

$$\neg x \vee ((\neg x \wedge \neg y) \vee x) = \neg x \rightarrow ((\neg x \wedge \neg y) \vee x) = 1 \in F.$$

$$x \vee (\neg x \wedge \neg y) = (x \vee \neg y) \wedge (x \vee y) = x \vee \neg y = y \rightarrow x \in F.$$

Now we are using the second hypothesis, $y \in F$:

$$y \rightarrow (x \wedge y) = \neg y \vee (x \wedge y) = (\neg y \vee x) \vee (\neg y \wedge x) = \neg y \vee x \in F,$$

so $x \wedge y \in F$.

Let $x \in F$ and $y \in B$. In the above computation we substitute $\neg y$ for y and we obtain $x \vee \neg(\neg y) \in F$, i.e. $x \vee y \in F$.

We can now come back to our concrete example of Boolean algebra, Lindenbaum algebra $P/_{\sim}$. Take $X \subset P$ and

$$F(X) = \{[a] \in P/_{\sim} | X \vdash a\}.$$

Then $F(X)$ is a filter. Indeed, $\mathbf{Taut} \in F(X)$ (if $a \in \mathbf{Taut}$, then $\models a$, so by the Completeness Theorem, $\vdash a$, so $X \vdash a$). Next take $[a] \in F(X)$ and $[a] \to [b] = [a \to b] \in F(X)$. This means that $X \vdash a, X \vdash a \to b$, so by *Modus Ponens*, $X \vdash b$, i.e. $[b] \in F(X)$.

So, from a structural point of view, $\{\mathbf{Taut}\}$ is a filter in the Lindenbaum algebra $P/_{\sim}$; actually, it is an ultrafilter.

Ultrafilters and Constructivity

Let B be a Boolean algebra. Filters in B which are maximal with respect to inclusion are called *ultrafilters*. It is not hard to show that *a filter F is an ultrafilter iff for every $x \in B$ either $x \in F$ or $\neg x \in F$, but not both.*

The key result on ultrafilters is the **Ultrafilter Theorem.** *Every filter in a Boolean algebra can be extended to an ultrafilter.*

To illustrate this situation we consider the Lindenbaum algebra $P/_{\sim}$; for every valuation h define

$$F_h = \{[a] \in P/_{\sim} | h(a) = 1\}.$$

A simple argument shows that F_h is a filter, in fact, an ultrafilter.

These facts actually motivate another approach to the Completeness Theorem, a path followed by Rasiowa and Sikorski. Bell and Slomson, p. 49, wrote:

> *The proofs of Post and Kalmár both provide explicit recipes for constructing a proof of a given tautology. The proof that we have given, which is due to Rasiowa and Sikorski, does not have this character, and since it depends on the ultrafilter theorem is not a constructive proof.*

This remark calls for a more detailed explanation. The problem under discussion is the following: Having a proposition a and *knowing* that a is a tautology, is it possible to get a **proof**, within the considered system, for a?

As it stands, the above question has *always* a positive answer, by the Completeness Theorem, independently of the proof of this theorem. Indeed, a dovetailing algorithm— a*British Museum algorithm*, following Chaitin—running through all possible proofs does the job, as *we know* that eventually the right **proof** will be discovered.

In fact, this is exactly the algorithm rejected for the general problem discussed at the beginning[a]. The main difference lies in the *extra* information given in the weaker question: we know that a is a tautology. This is a quite subtle situation, in which the difference between a constructive proof and a non-constructive proof affects very little the "numerical" content of the result.

[a] Having a proposition a, is it possible to algorithmically decide if a is a **theorem**?

To get more insight on this phenomenon we make use of some rudiments of Constructive Mathematics. According to Bridges and Richman:

> *We engage in constructive mathematics from a desire to clarify the meaning of mathematical terminology and practice – in particular, the meaning of existence in a mathematical context. The classical mathematician, with the freedom of methodology advocated by Hilbert, perceives an object x to exist if he can prove the impossibility of its nonexistence; the constructive mathematician must be presented with an algorithm that constructs the object x before he will recognize that x exists.*

The essential difference between a classical and constructive approach to mathematics can be grasped by considering **binary sequences** generated by an algorithm. Let $b = b_1 b_2 \ldots b_n \ldots$ be a binary sequence and consider the following statements:

$$S(b) : b_n = 1, \text{ for some } n,$$

$$\neg S(b) : b_n = 0, \text{ for all } n.$$

Here $\neg S(b)$ is the denial of $S(b)$: under the assumption of $S(b)$ a contradictory statement (like $0 = 1$) holds. A constructive proof of $S(b) \vee \neg S(b)$ must provide an algorithm showing that $b_n = 0$, for all n, or computing a positive integer n such that $b_n = 1$.

A **Brouwerian counterexample** to an assertion is a proof that the assertion implies some unacceptable principle in constructive mathematics. The most popular such principle is called the **Limited Principle of Omniscience, LPO**:

> *If (b_n) is a binary sequence, then either there exists n such that $b_n = 1$, or else $b_n = 0$, for all n.*

Clearly, **LPO** is simple the assertion

$$\forall b(S(b) \vee \neg S(b)).$$

Why is it constructively false? Just because it is equivalent to the **Halting Problem**, which cannot be solved algorithmically. Here is the outline of the argument. First we show that the **Halting Problem** is not decidable. Assume, for the sake of a contradiction, that there exists a **halting program** deciding if an arbitrary program eventually halts. The outputs, if any, for all our programs are supposed to be 0 or 1; also, we may assume, without loss of generality, that the inputs for the programs are part of the programs themselves.

- read a natural N;
- generate all programs up to N bits in size;
- use the **halting program** to check for each generated program whether it halts and filter out all non-halting programs;
- simulate the running of the above generated programs;
- make sure that the running time of the current program is bigger than the running time of all halting programs, generated above.

First, notice that the above program eventually halts for every natural N. How long is the above program ? It is about $\log_2 N$ bits. Indeed, the program consists of the input data N (which requires about $\log_2 N$ bits) and a constant part. Globally, the program has $\log_2 N + O(1)$ bits. For large enough N, the above program will belong to the set of programs having less than N bits (because $\log_2 N + \mathrm{O}(1) < N$). Accordingly, the program will be generated by itself – at some stage of the computation. In this case we have got a contradiction, since the computation time for our program will be bigger than the computation time of itself!

The second step in our argument is a reduction: we prove that in case we assume **LPO**, then the **Halting Problem** is decidable. Indeed, let

$$\pi_1, \pi_2, \ldots, \pi_n, \ldots$$

be the set of all our programs and consider the following (computable) function μ:

$$\mu(m,k) = \begin{cases} 1, & \text{if } \pi_m(m) \text{ halts in time less than } k, \\ 0, & \text{otherwise.} \end{cases}$$

Applying **LPO** to the set of binary sequences

$$b_1^m, b_2^m, \ldots, b_i^m, \ldots,$$

where

$$b_i^m = \mu(m, i),$$

would solve the **Halting Problem**, which is impossible.

Markov's Principle, MP—this principle is rejected by Brouwer, but freely used by the Russian school in constructive mathematics—which corresponds to our axiom A3., reads:

$$\neg\neg S(b) \Leftrightarrow S(b).$$

To illustrate **MP** we consider an one-one enumeration of all possible proofs for tautologies:

$$p_1, p_2, \ldots$$

Let $a \in$ **Taut** and consider the predicate

$$Pred(a, i) = p_i \text{ is a proof for } a.$$

Clearly, for a fixed a, $Pred(a, i)$ is algorithmically decidable. Next, use **MP** to the statement

$$\exists i Pred(a, i),$$

saying that *it would be absurd to deny that there is a positive integer i such that $Pred(a, i)$.* From this fact we get no clue or computation bound for the construction of such an i.

Note, following Brouwer, that the statement

$$\neg\neg\neg A \Rightarrow \neg A,$$

is constructively meaningful. Indeed, under the assumption $\neg\neg\neg A$ we can prove $\neg a$ by deriving a contradiction from A: if A, then $\neg A$ is absurd, hence $\neg\neg A$, which contradicts $\neg\neg\neg A$. To conclude, given a tautology a, we surely can get a **proof** for a; however, we can provide no indication concerning the number of proofs necessary to inspect before getting the required **proof**: this state of affairs reflects the meaning of **MP**.

Decidability and Complexity

The formulation of the negative result cited at the end of the above section leaves the impression that the property of being a **theorem** of the propositional calculus is algorithmically decidable. Is this true?

Indeed, the above decision problem is algorithmically decidable, by virtue of the Completeness Theorem. **Theorems** coincide with tautologies and testing if an arbitrary proposition is or is not a tautology is algorithmically decidable.

Apparently, switching from **theorems** to tautologies doesn't help too much, as we replace the (potential infinite) search through all possible proofs by a search through all possible valuations of propositions (an infinite set, as well). This is only a superficial feeling! If a contains n atomic propositions, then we don't have to go through all possible valuations h, but to examine only the restriction of these valuations to the set of atomic propositions in a. We have arrived at a finite set, containing 2^n elements.

Question is: How difficult is to decide if an arbitrary proposition a is a **theorem**? To approach this question we will discuss briefly the class **P** of polynomial algorithms. The Euclidean algorithm (for computing the greatest common divisor of two positive integers) is an example of a polynomial algorithm, in the sense that the number of basic bit operations (the *time* used by the algorithm) is a polynomial function of the number of bits in the input.

If the two inputs have at most n digits, then the larger input is less than 2^n, so the number of steps is bounded by cn. Each step involves a division, i.e. about n^2 bit operations, which gives finally a cubic polynomial. In fact, by a more involved analysis one can show that the Euclidean algorithm works in time bounded by a quadratic function of the length of inputs.

Sometimes we are not able to design a polynomial algorithm for a problem; at least, we can prove that a solution for the problem can be quickly recognized if some miraculously source furnishes it to us. A good example is the problem of primality. It is not easy to determine if a positive integer having 1,000 digits is composite. But if we have got somehow two numbers and a claim that they multiply to the given number, then we can very easily check if the claim is correct or not. These two numbers form a *certificate* of their compositeness. This leads to the important class **NP** of algorithms running in nondeterministic polynomial time, i.e. algorithms working in polynomial time under the assumption that a certain certificate has been given. One of the most intriguing open problems in theoretical computer science pertains exactly the relation between **P** and **NP**: **P** =? **NP**.

Testing if an arbitrary proposition a is a **theorem** along the path suggested at the beginning of this section requires an exponential computation time. Is it possible to do it better? No one knows this. We can see readily that our problem is in co-**NP**, as guessing an valuation which makes a false solves very quickly the problem.

In fact the problem $\mathbf{P} = ?\ \mathbf{NP}$ is really meta-mathematical! Indeed, assume an appropriate coding and measure of the size of proofs. So, we may have polynomial size proofs and exponential size proofs. The difference between $\mathbf{P}$ and $\mathbf{NP}$ – if any – may be seen as a difference between *constructing* a polynomial size proof and *verifying* a polynomial size proof. If $\mathbf{P} = \mathbf{NP}$, then they are the same.